UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/662,588	09/15/2000	Fred Irwin	CITI0184	1954

27510 7590 07/09/2007
KILPATRICK STOCKTON LLP
607 14TH STREET, N.W.
WASHINGTON, DC 20005

EXAMINER	
BORLINGHAUS, JASON M	
ART UNIT	PAPER NUMBER
3693	

MAIL DATE	DELIVERY MODE
07/09/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 09/662,588
Filing Date: September 15, 2000
Appellant(s): IRWIN ET AL.

George Marcou
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 5/09/07 appealing from the Office action mailed 12/19/05.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

4,674,044 Kalmus et al. 01-1985

Schildt, Herbert. Turbo C/C++: The Complete Reference. Osbourne McGraw-Hill. Berkeley, CA . (1990) pp. 13, 561 and 727-720.

Coughlin, George Gordon. Your Handbook of Everyday Law: 5th Edition. Harper Collins Publishing. New York, NY. (1993). pp. 50 - 51.

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.

2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

Claims 8 – 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kalmus (US Patent 4,674,044) in view of Schlidt (Schildt, Herbert. *Turbo C/C++: The Complete Reference. Osborne McGraw-Hill.* Berkeley, CA. 1990. pp. 13, 561 and 727 - 730) and (Coughlin, George Gordon. *Your Handbook Of Everyday Law. 5th Edition. Harper Collins Publishing.* New York, NY. 1993. pp. 50 – 51).

Regarding Claim 8, Kalmus discloses a system comprising:

- a customer terminal (brokerage firm's account executives and/or computer equipped customers). (see col. 4, lines 60 – 69);
- a trader terminal (see 15, figure 1) operatively coupled to the customer terminal (see 27 and 29, figure 1) through a computer network (11, 25 and 26, figure 1);
- a processor (CPU – see 10, figure 1);
- wherein the processor (CPU) is involved in a trade request (order generated) from a customer at a customer terminal (brokerage firm's account executives and/or computer equipped customers - see col. 4, lines 60 – 69) further comprising:
- a first component comprising functions for sending messages and receiving messages to the system (CPU – see 10, figure 1) on behalf of the customer (Brokerage House – see 27, figure 1). ("Input/output network 25 provides data communication with the various branch offices 27 of the

brokerage house. Line 25 permits communication with either the branch order entry clerk or directly to the account executives at each branch." – see col. 4, lines 41 – 45 - It is inherent that a component comprises the functions for sending messages and receiving messages to the system on behalf of a customer.);

- a second component comprising functions for controlling access to the system by the customer. ("The order data fields include ... customer identification (CUSTID)..." – see col. 5, lines 52 – 54 – It is inherent that a component comprises functions for controlling access to the system by the customer since the functioning of the system requires a valid customer identification); and
- a third component comprising functions for sending messages to and receiving messages from the first component and a trader (Trader – see 15, figure 1) at the trader terminal. (see 11, figure 1 - It is inherent that a component comprises functions for sending messages to and receiving messages from the first component and a trader.)

Kalmus does not teach a system comprising:

- a processor is configured to dynamically create sets of class components to handle one or more transactions with each set of class components; and

- wherein the processor comprises a timer wherein the trade request from the customer is automatically revoked at a predetermined duration of time if the trader does not accept the trade request.

Object-oriented programming and class-based programming are old and well known in the art for designing, modeling, building and developing software systems, as evidenced by Schildt who states that object-oriented programming and class-based programming, an object-based programming language, has been in use since 1980 ("In 1980, while working at Bell Laboratories at Murray Hill, New Jersey, Bjarne Stroustrup addressed this problem by adding several extensions to the C language. Initially, called 'C with Classes,' the name was changed to C++ in 1983." – see page 727). It would have been obvious to a person of ordinary skill in the art to have modified Kalmus to allow for his trading system to be encoded or programmed in any computer language that the inventor desired, including object-oriented programming language or class-based programming language which were already in use at the time of invention.

Furthermore, an object-based programming language and a class-based programming language, which were both known at the time of the invention, would have been obvious choices by which to reinterpret Kalmus due to the numerous benefits of such programming languages – easy programming maintenance, easy to understand and streamlined structure, as evidenced by Schildt who states, "Object-oriented programming allows you to easily decompose a problem into subgroups of related parts. Then you translate these subgroups into self-contained units called objects" (see

page 729) and "Also, because C++ shares C's efficiency, high-performance systems software can be constructed using C++." (see page 728).

Setting a time limit for acceptance of an offer and the revocation of said offer if not accepted within the established time limit is old and well known in the art of contract formation, as evidenced by Coughlin who states "For how long is an offer good, and how is an offer terminated? Generally, an offer is deemed to be terminated ... (2) by the lapse of the time specified or the lapse of a reasonable time when the offer is silent concerning duration." (see Termination of Offer, p. 50). It would have been obvious to one of ordinary skill to have modified Kalmus and Schildt by incorporating into the processor a timer which would revoke the trade request (offer) when such trade request (offer) was not accepted within the predetermined duration of time, as is old and well known, to account for time limitations on trade requests (offers) sent to traders.

Regarding Claims 9 – 14, Kalmus discloses a system wherein:

- the third component operates in a synchronous (real-time) format. ("The market making system of the above-described invention has thus been shown to automatically accommodate a random, real time order flow for security purchases or sales." – see col. 10, lines 23 – 26);
- the third component operates in a asynchronous (time delay executability) format. ("Orders not qualified for execution are stored and re-examined from time to time for possible later executability." – see col. 10, lines 34 – 36); and

- components are configured to handle multiple customers at one time.("While only one branch is shown in Figure 1, it is to be understood that a multiplicity of branches 27 are in data communication with processor 10." – see col. 4, lines 45 – 48 – It is inherent that the set of class components are configured to handle multiple customers at one time).
- components are configured to handle multiple transactions at one time. ("While only one branch is shown in Figure 1, it is to be understood that a multiplicity of branches 27 are in data communication with processor 10." – see col. 4, lines 45 – 48 – It is inherent that the components are configured to handle multiple transactions at one time with multiple customers in data communication with the processor.).

Kalmus does not teach a system wherein:

- the set of class components are configured to handle multiple customers at one time;
- the set of class components are configured to handle a single customer at one time;
- the set of class components are configured to handle a single transaction at one time; and
- the set of class components are configured to handle multiple transactions at one time.

However, functionality can be deleted from Kalmus to reduce the its ability to handle multiple customers and multiple transactions at one time:

- to handle a single customer at one time; and
- to handle a single transaction at one time.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to have modified Kalmus by incorporating the ability to handle both multiple and single customers, and both multiple and single transactions, to allow for versatility in the operation and functioning of the system.

Additionally, as discussed above in Claim 8 and is evidenced by Schildt, object-oriented programming and class-based programming are old and well known in the art for designing, modeling, building and developing software systems.

Accordingly, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to have modified Kalmus to allow for his trading system to be encoded or programmed in any computer language that the inventor desired, including object-oriented programming language or class-based programming language which were already in use at the time of invention was made, to reap the benefits of that computer programming language, as discussed above.

Regarding Claim 15, Kalmus does not teach a system wherein:

- the processor creates sets of class components based on the number of transactions.

However, as discussed above in Claim 8 and is evidenced by Schildt, object-oriented programming and class-based programming are old and well known in the art for designing, modeling, building and developing software systems.

Therefore, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to have modified Kalmus to allow for his trading system to be encoded or programmed in any computer language that the inventor desired, including object-oriented programming language or class-based programming language which were already in use at the time of invention was made, to reap the benefits of that computer programming language, as discussed above.

Additionally, it is old and well known in the art that software creates and assigns memory space (an object, a memory location, a file) to store customer information and customer transaction data during customer login and transaction execution. As evidenced by Schildt who states memory is allocated during transaction execution "In this method, storage for information is allocated from the free memory area as it is needed and returned to free memory when it has served its purpose...Because memory can be allocated for one purpose and freed when that use has ended, it is possible for another part of the program to use the same memory for something else at a different time." (see page 561).

Therefore, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to have modified Kalmus to allow for the processor to create class components based on the number of transactions since, as evidenced by Schildt,

memory space would need to be allocated based on the number of transactions to be executed by the system, as is standard in software programming.

Regarding Claim 16, further method claim would have been obvious from system claim rejected above, Claim 8, and is therefore rejected using the same art and rationale.

Regarding Claim 17, Kalmus does not teach a method wherein:

- each component is created in response to a customer accessing the system.

It is old and well known in the art that software creates and assigns memory space (an object, a memory location, a file) to store customer information and customer transaction data during customer login and transaction execution. As evidenced by Schildt who states memory is allocated during transaction execution "In this method, storage for information is allocated from the free memory area as it is needed and returned to free memory when it has served its purpose...Because memory can be allocated for one purpose and freed when that use has ended, it is possible for another part of the program to use the same memory for something else at a different time." (see page 561).

Therefore, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to have modified Kalmus to create components in response to a customer accessing the system since, as evidenced by Schildt, memory space would need to be allocated based on system access by customers, as is standard in software programming.

Regarding Claim 18, Kalmus discloses a computer program comprising:

- at least one computer-readable medium. (It is inherent that the system would require software programming contained on a computer-readable medium to function); and
- a programming module stored on the at least one medium, and operable, upon access of a customer to trading services of the computer program product for handling one or more transactions involving a trade request from the customer to a trader (""The order data fields include ... customer identification (CUSTID)..." – see col. 5, lines 52 – 54 – It is inherent that a valid customer identification is required to allow access to trading services) to;
- where created programming include at least one of:
  - an access control programming. ("The order data fields include ... customer identification (CUSTID)..." – see col. 5, lines 52 – 54 – It is inherent that an access control programming must exist in order to validate customer identification);
  - a trading system communications programming. (see 11, 12, 22, 23, 25 and 26, figure 1 - It is inherent that a trading system communications programming must exist to manage communications for the trading system); and
  - a translator programming ("...data input apparatus such as a keyboard..." – see col. 4, lines 9 – 10 – It is inherent that a translator

programming must exist to convert customer input into executable transactions.)

Kalmus does not teach a computer program comprising:

- a class creation module stored on the at least one medium, and operable, upon access of a customer to trading services of the computer program product, to create at least one set of classes, each set comprising at least one class;
- where created classes include at least one of:
  - an access control class;
  - a trading system communications class; and
  - a translator class; and
- a timer module stored on the at least one medium; and
- operable to automatically revoke at a predetermined time the trade request from the customer if the trader does not accept the trade request.

However, as discussed above in Claim 8 and is evidenced by Schildt, object-oriented programming and class-based programming are old and well known in the art for designing, modeling, building and developing software systems.

Therefore, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to have modified Kalmus to allow for his trading system to be encoded or programmed in any computer language that the inventor desired, including object-oriented programming language or class-based programming language which

were already in use at the time of invention was made, to reap the benefits of that computer programming language, as discussed above.

Furthermore, as discussed above in Claim 8 and is evidenced by Coughlin, setting a time limit for acceptance of an offer and the revocation of said offer if not accepted within the established time limit is old and well known in the art of contract formation.

Therefore, it would have been obvious to one of ordinary skill to have modified Kalmus and Schildt by incorporating into the processor a timer which would revoke the trade request (offer) when such trade request (offer) was not accepted within the predetermined duration of time, as is old and well known, to account for time limitations on trade requests (offers) sent to traders.

Regarding Claims 19 – 20, further computer program claims would have been obvious from system claim rejected above, Claims 13- 14, and are therefore rejected using the same art and rationale.

Regarding Claim 21, Kalmus does not teach a computer program where:

- each class being an object linking and embedded class type.

However, as discussed above in Claim 8 and is evidenced by Schildt, object-oriented programming and class-based programming are old and well known in the art for designing, modeling, building and developing software systems. Furthermore, object-linking and embedded class types are standard in object-oriented programming and class-based programming languages. As evidenced by Schildt that discusses

object linking by stating "Later the linker combines the code you wrote with the object code already found in the standard library. This process is called linking." (see page 13).

Therefore, at the time of the invention, it would have been obvious to a person of ordinary skill in the art to have modified Kalmus to allow for his trading system to be encoded or programmed in any computer language that the inventor desired, including object-oriented programming language or class-based programming language which were already in use at the time of invention was made, and utilizing all techniques commonly employed by that programming language, to reap the benefits of that computer programming language, as discussed above.

Furthermore, as discussed above in Claim 8 and is evidenced by Coughlin, object-oriented programming and class-based programming are old and well known in the art for designing, modeling, building and developing software systems. Furthermore, object-linking and embedded class types are standard in object-oriented programming and class-based programming languages. As evidenced by Schildt that discusses object linking by stating "Later the linker combines the code you wrote with the object code already found in the standard library. This process is called linking." (see page 13).

Regarding Claim 22, further computer program claims would have been obvious from a subsection of the program claim rejected above, Claim 18, and is therefore rejected using the same art and rationale.

Regarding Claim 23, further method claim would have been obvious from a subsection of the program claim rejected above, Claim 18, and is therefore rejected using the same art and rationale.

Regarding Claims 24 - 25, further method claims would have been obvious from system claim rejected above, Claims 13- 14, and are therefore rejected using the same art and rationale.

Regarding Claim 26, further method claim would have been obvious from computer claim rejected above, Claim 21, and is therefore rejected using the same art and rationale.

Regarding Claim 27, further method claim would have been obvious from a subsection of the program claim rejected above, Claim 18, and is therefore rejected using the same art and rationale.

(10) Response to Argument

Analogous Art

As a preliminary matter, Examiner asserts that the prior art references (Kalmus, Schildt and Coughlin) are valid under the analogous arts test. The Courts have stated that to be utilized "as a basis for rejection of the applicant's invention, the reference must either be in the field of the applicant's endeavor or, if not, then be reasonably pertinent to the particular problem with which the inventor was concerned." *In re Oetiker, 977 F.2d 1443, 1447 (Fed. Cir. 1992).* As such "it is necessary to consider "the reality of the circumstances" -- in other words, common sense -- in deciding in which fields a person of ordinary skill would reasonably be expected to look for a solution to the problem facing the inventor." *In re Wood, 599 F.2d 1032, 1036 (CCPA 1979).*

Examiner asserts that based upon common sense, the field of the references and/or the problem the inventor was concerned about, that the cited prior art references would have been utilized by a skilled artisan in the art.

Kalmus discloses a computerized securities trading system that retrieves, matches and executes security trades based upon transmitted bids and offers. Schildt discloses a standard and conventional programming language utilized to encode computerized systems, while Coughlin discloses standard and conventional legal practices related to offers to buy and sell goods. While Schildt and Coughlin do not directly relate to securities trading, Examiner asserts that such does not disqualify their applicability as to their teachings, as the motivation to consult such references for their teachings should be readily apparent.

Claim 8

In response to Appellant's argument that prior art reference(s), neither alone nor in combination, discloses nor suggests the claim limitation of Claim 8 of "wherein the processor comprises a timer wherein the trade request from the customer is automatically revoked at a predetermined duration of time if the trader does not accept the trade request", the Examiner refutes such an assertion.

Kalmus discloses:

> ***The processor 10 first determines whether or not each received order can be executed, i.e., "qualifies" the order***. There are various reasons why an order will not be executed by the market maker. Thus, for example, the customer may seek to sell stock above the current bid price or to purchase the security below the current asked price. A customer may seek to trade a number of shares which exceeds the amount which the particular market maker is willing to accommodate, either in gross or for any one order. ***Orders not executable, i.e.,***

> ***orders not qualified, are either stored in memory in the processor 10 for later execution if they become qualified*** (such as by a favorable change in the market price for a security which can then accommodate the customer's price limits) or are forwarded to other market makers for potential execution over communication links 23 or 26. (emphasis added – see col. 5, lines 6 – 21).

Therefore, according to Kalmus, the processor qualifies a received securities order, determining its executability. If said order cannot be executed then storing said order in the processor until such a time as said order can be executed.

Coughlin, which discloses general legal information concerning the issuance, acceptance and revocation of offers, states:

> For how long is an offer good, and how is an offer terminated? Generally, an offer is deemed to be terminated (1) when it has been rejected by the person to whom it was made; (2) by the lapse of the time specified or the lapse of a reasonable time when the offer is silent concerning duration. (see p. 50).

Coughlin then cites numerous examples in which the validity of an extended offer was conditioned based upon its acceptance by a specified time or date.

Examiner asserts that these two references, Kalmus and Coughlin, when read in combination do disclose the contested claim limitation. As demonstrated by Coughlin, offers have either a specified time period for acceptance before their expiration or an innate reasonable time period for acceptance before their expiration. According to Kalmus, the processor qualifies whether an offer can be executed. Examiner asserts that it would have been obvious to one of ordinary skill in the art at the time the invention was made to have modified the processor of Kalmus to qualify submitted offers based upon their time-based validity, as disclosed by Coughlin.

Examiner would like to point out that, according to Kalmus, should an order be deemed not executable, said order is stored in memory for later execution. However,

without the time-based modification, an order submitted to the system and deemed unqualified, such as due to price and volume, could be stored indefinitely. The user of such a system could find themselves liable for a stored order that was finally executed days, months or years after it had been originally submitted.

Claims 16 and 18

All argument(s) and/or rationale(s) set forth above with respect to earlier addressed claim(s), Claim(s) 8, are hereby incorporated and/or reapplied so as to apply to Claim(s) 16 and 18 where applicable.

Examiner would like to additionally point out that while Claim 8 claims "a timer" and Claim 18 "a timer module stored on one medium" which would have been obvious based upon automation of the above cited references, Claim 16 merely claims "automatically revoking at a predetermined duration of time the trade request from the customer if the trade has not accepted the trade request which reads directly on the cited portion of Coughlin.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Jason Borlinghaus (JMB)

June 21, 2007

Conferees:

James Kramer

ANDREW J. FISCHER
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 3600